IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES
(Attorney Docket No. 003797.00212)

In re U.S. Patent Application of Giovanni M. Della-Libera	)	
Application No. 10/068,444	)	Confirmation No. 9546
Filed: February 6, 2002	)	Group Art Unit: 2132
For: Virtual Distributed Security System	)	Examiner: Farid Homayounmehr

BRIEF ON APPEAL

Mail Stop: Appeal Brief-Patents
Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450

This is an Supplemental Appeal Brief in accordance with 37 CFR §1.192 filed in support of Applicant's Notice of Appeal. Appeal is taken from the Non-Final Office Action dated October 13, 2006. Please charge any necessary fees in connection with this appeal brief to our Deposit Account No. 19-0733.

I. REAL PARTY IN INTEREST

The owner of this application, and the real party in interest, is Microsoft Corporation.

II. RELATED APPEALS AND INTERFERENCES

U.S. App. 11/254,519 ("the '519 application") and U.S. App. 11/254,264 ("the '264 application") were filed on October 20, 2005 and are divisional applications of the present application. As set forth in the Appeal Brief dated January 12, 2007, Pre-Appeal Request for Reviews were filed concurrently with a Notice of Appeal for both the '519 application and the '264 application on November 13, 2006. As of today, prosecution has been reopened in the '264 application and the '519 application remains under appeal.

III. STATUS OF CLAIMS

Claims 1 – 21, 33 and 34 remain in the application. Claims 22-32 were withdrawn from consideration. All of the pending claims are shown in the attached appendix.

IV. STATUS OF AMENDMENTS

There are no amendments subsequent to the Non-Final Office Action dated October 13, 2006.

V. SUMMARY OF CLAIMED SUBJECT MATTER

In making reference herein to various portions of the specification and drawings in order to explain the claimed invention (as required by 37 CFR §41.37(c)(1)(v)), Applicant does not intend to limit the claims. All references to the specification and drawings are illustrative unless otherwise explicitly stated.

The claimed subject matter is directed towards providing security in distributed computer systems. (Paragraph 02, lines 2-3). Specifically, the claimed systems and methods utilize a security protocol independent framework that can be scaled for use with wide area networks, such as the Internet, and that is independent of the underlying cryptographic mechanisms being used. (Paragraph 05, lines 2-4). By utilizing a security policy written in a security protocol independent security policy language, the underlying protocols are abstracted, thus allowing the security runtime to support different platforms, technologies, and protocols. (Paragraph 6, lines 13-16; Figures 8-10 show exemplary security policies).

The security protocol independent security framework as recited, and explained in more detail below, can support multiple cryptographic technologies and by abstracting the technologies "does not require applications to be aware of the cryptographic technologies being used". (paragraphs 53, lines 9-11). Moreover, as indicated in paragraph 43 of the present application, abstracting underlying protocols facilitates interoperability with other systems. Thus, the distributed security system as currently recited can incorporate a security policy that may be simultenouesly implemented across different platforms that support different protocols and cryptographic techniques.

Independent claim 1 is directed to a distributed security system comprising "a security policy written in a security protocol independent security policy language." As discussed above, by utilizing a security policy written in a security protocol independent security policy language, the underlying protocols are abstracted, thus allowing the security runtime to support different platforms, technologies, and protocols. (Paragraph 6, lines 13-16). Claim 1 further recites this aspect by indicating "the security policy is configurable to be simultaneously implemented for a plurality of computer devices within the distributed security system, wherein at least a first computer device within the distributed security system operates on an operating platform that supports at least one security protocol that is different than a security protocol supported by a

platform of at least a second computer device among the plurality of computer devices." Figure 6 shows an application communicating with an existing security component using an exemplary distributed security system. As explained in relation to Figure 6, "[i]nteroperability can occur because virtual distributed security 602 may select the most appropriate protocol 610 and transport 612 based on information included in a security policy 608." (Paragraph 43, lines 4-5). The claim concludes by stating "the first and the second computer devices process data in accordance with the security policy of the distributed security system."

Independent claim 33 is directed towards a method for utilizing a security policy. Claim 33 comprises two elements. The first element recites:

> implementing a security policy written in a security protocol independent security policy language within a distributed computing system, wherein the distributed computing system comprises at least a first computer device operating on a first operating platform and at least a second computer device operating on a second operating platform that supports at least one security protocol that is different than a security protocol supported by a platform of the first computer device

As discussed above, by utilizing a security policy written in a security protocol independent security policy language, the underlying protocols are abstracted, thus allowing the security runtime to support different platforms, technologies, and protocols. (Paragraph 6, lines 13-16). As one example provided in the Specification:

> In prior art systems, the rights that may be utilized are hard coded within the security system. Windows NT operating systems has 32 defined permission rights. With the present invention, the administrator can define new rights by defining or editing a security policy. The secuirty policy may be capability based, i.e, an application may define a capability and virtual distributed security system 202 may provide that capability.

(Paragraph 27, lines 5-11). The second step of independent claim 33 is "configuring the security policy to allow the first computer device and the second computer device to simultaneously process data in accordance with the security policy of the distributed security system." Figure 1 illustrates a distributed computing system operating environment. As seen, computer device 104, "[c]omputer device 106 and computer device 108 may be coupled to communications network 102 through communication devices...[and] may communicate with one another via communication network 102." (See paragraph 23, lines 1-2 and 5-7). Figure 2 shows the

architecture of a virtual distributed security system which may be implemented the computing system operating environment shown in Figure 1.

Independent claim 34 is directed towards a computer readable medium having computer-executable instructions for utilizing a security policy. Specifically, when executed, the computer-executble instructions apply "a security policy within a distributed computing system having at least a first computer device operating on a first operating platform and at least a second computer device operating on a second operating platform that supports at least one security protocol that is different than a security protocol supported by a platform of the first computer device, wherein the security policy is written in a security protocol independent security policy language." Figure 1 illustrates a distributed computing system operating environment. As seen, computer device 104, "[c]omputer device 106 and computer device 108 may be coupled to communications network 102 through communication devices...[and] may communicate with one another via communication network 102." (See paragraph 23, lines 1-2 and 5-7). Figure 2 shows the architecture of a virtual distributed security system which may be implemented the computing system operating environment shown in Figure 1. By utilizing a security policy written in a security protocol independent security policy language, the underlying protocols are abstracted, thus allowing the security runtime to support different platforms, technologies, and protocols. (Paragraph 6, lines 13-16; Figures 8-10 show exemplary security policies).

The second step recites "permitting the security policy to be simultaneously implemented for a plurality of computer devices within the distributed security system including at least the first computer device and the second computer device." As explained in relation to Figure 6, "[i]nteroperability can occur because virtual distributed security 602 may select the most appropriate protocol 610 and transport 612 based on information included in a security policy 608." (Paragraph 43, lines 4-5).

Thus, as recited in each of the independent claims (claims 1, 33, and 34), the distributed security system as currently recited can incorporate a security policy that may be simultenouesly implemented across different platforms that support different protocols and cryptographic techniques.

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1) Claims 1, 2, 3 and 5-19, 32 and 34 are rejected under 35 U.S.C. §102(b) as allegedly being anticipated by Rothermel (US Patent No. 6,678,827).

2) Claims 4, 20 and 21 are rejected under 35 U.S.C. §103(a) as allegedly being unpatentable over Rothermel (US Patent No. 6,678,827) as applied to claim 1, and further in view of Saulpaugh (US Patent No. 6,850,979).

ARGUMENT

A. Rothermel Does Not Teach, Disclose or Suggest the Claim Limitations

Rothermel teaches distributing a consistent security policy template to network security devices (See Col. 3, lines 33-34). While the consistent template may be configured with specific information, Rothermel does not teach or even suggest using a security protocol independent security policy language to create such a policy template or the policy itself. The security policy templates of Rothermel must use the existing security protocols utilized by the network security devices, thus are not equivalent to the security policy written in a security protocol independent security policy language of the independent claims 1, 33 and 34.

As indicated in paragraph 6 and recited in the claims 1, 33 and 34 of the present application, a security framework that is security protocol independent can support multiple cryptographic technologies across different platforms. As discussed above, by utilizing a security policy written in a security protocol independent security policy language as recited in the pending claims, the underlying protocols are abstracted, thus allowing the security runtime to support different platforms, technologies, and protocols. This is in stark contrast to the systems disclosed in Rothermel, which does not teach, disclose or suggest simultaneous implementation across different platforms or a security policy written in a security protocol independent security policy language.

The Examiner asserts that Col. 13, line 30 to Col. 14, line 13 of Rothermel demonstrates a system having interoperability with multiple OSs, and therefore must be utilizing a security policy written in a security protocol independent security policy language. (See Advisory Action dated 06/08/2006, page 2 and Office Action dated 03/12/006, page 4). The Applicants respectfully disagree with such an interpretation since the cited text explicitly states:

> In the illustrated embodiment, the NSD is a security appliance device capable of executing the Linux operating system...The NSD software components include a version of the Linux OS kernel 610 which is capable of executing on the NSD to provide various OS functionality (e.g., TCP/IP support, network drivers, etc.).

(Col. 13, lines 33 – 43; emphasis added). Indeed, the cited text further discusses software components "which interacts directly with the OS" (emphasis added), such as the packet filter engine, the firewall component, and functionality-specific drivers (e.g., VPN drivers). Therefore, since all the devices are managed in conjunction with the LINUX OS, there is no use or

motivation to even consider applying a security policy written in a security protocol independent security policy language. Thus, Applicants respectfully disagree with the Examiner assertion that "[a] variety of optional software components can be provided to and executed by an NSD (column 14, lines 31-33). Therefore, Rothermel suggests simultaneous configuration of different NSDs running operational platforms." (Advisory Action dated October 13, 2006, page 3). As discussed above, the referenced NSD is a security appliance device capable of executing the Linux operating system, where the various software components may provide various OS functionality.

The Applicants further note that Rothermel later mentions "various specific types of software (e.g., the Linux OS and the TCP/IP protocol) could be replaced with alternate types of software providing similar functionality." (Col. 14, lines 42-45). As stated, the LINUX OS could be replaced, however there is no specifically no mention or suggestion to combine the use of other OSs or otherwise amend the current setup discussed in relation to Fig. 6 of Rothermel for which the Office Action cites. Therefore, there is no disclosure or suggestion of simultaneous implementation across different platforms or a security policy written in a security protocol independent security policy language.

In fact, Rothermal merely discloses a system for managing multiple related network security devices with a security policy template. Rothermel never states or even suggests that the multiple security devices assigned to a specific supervisor device utilize a security policy written in a security protocol independent language. Rather, a copy of a security policy template is sent to related network security devices from a supervisor device. There is no specific teaching or suggestion that the template is even written in a security protocol independent language, rather that the template is tailored towards specific devices. The Applicants cannot locate any disclaimer to this interpretation.

The Applicants further disagree with the allegation that Col. 7, lines 3 – 57 shows a security policy written in a security protocol independent language. The relevant portion of the cited text states:

> When a user of the manager device desires to establish or modify a security policy for one or more NSDs such as NSDs 130 and 140, the user first selects one of the security policy templates 113 or creates a new security policy template. Security policy templates are discussed in greater detail below with respect to FIG. 3. The manager device then determines the one or more primary supervisor devices for

> the NSDs of interest, such as by retrieving this information from its specific security policy information 116. If this information is not stored by the manager device, the manager device can obtain the information in a variety of ways, such as by querying the NSDs of interest or by querying the various known supervisor devices.

(emphasis added). As explained throughout the text of Rothermel, a copy of a specific security policy template is sent to related network security devices from a supervisor device. If the systems of Rothermel could utilize a security protocol independent language, there would be no reason to query specific supervisor devices or otherwise determine appropriate supervisor devices, because any of them could be utilized. For example, as seen in Figure 1 and explained in Col. 6, only certain network security devices are in communication with certain supervisor devices.

For at least these reasons, Applicants respectfully submit that claims 1, 33 and 34, along with their respective dependent claims are in condition for allowance, and therefore, respectfully request reversal of the pending rejections utilizing Rothermel.

CONCLUSION

The rejections contained in the Action of October 13, 2006 should be reversed for at least the reasons recited above. Reversal of the rejections is requested.

Respectfully submitted,

Date: April 9, 2007

SHAWN P. GORMAN
REG.# 56,197
BANNER & WITCOFF, LTD.
10 S. Wacker Drive
Suite 3000
Chicago, IL 60606-7407
Telephone: 312-463-5000
Facsimile: 312-463-5001

CLAIMS APPENDIX

1. (Previously Presented) A distributed security system comprising:

a security policy written in a security protocol independent security policy language, wherein the security policy is configurable to be simultaneously implemented for a plurality of computer devices within the distributed security system, wherein at least a first computer device within the distributed security system operates on an operating platform that supports at least one security protocol that is different than a security protocol supported by a platform of at least a second computer device among the plurality of computer devices wherein the first and the second computer devices process data in accordance with the security policy of the distributed security system.

2. (Original) The distributed security system of claim 1, wherein:

the security policy identifies components of the security system.

3. (Original) The distributed security system of claim 1, wherein:

the security policy identifies access rights of the security system.

4. (Original) The distributed security system of claim 1, wherein:

the security policy language comprises the extensible markup language.

5. (Original) The distributed security system of claim 1, wherein:

the security policy is configurable.

6. (Original) The distributed security system of claim 1, wherein:

the security policy language comprises at least some logic-based components.

7. (Original) The distributed security system of claim 1, wherein:

the security policy language comprises at least some rule-based components.

8. (Original) The distributed security system of claim 1, wherein:

the security policy language comprises procedural components.

9. (Original) The distributed security system of claim 1, wherein the computer device is configured with computer-executable instructions to:

receive from a first entity a message formatted in a first protocol; and

transmit to a second entity the message formatted in a second protocol that is different from the first protocol.

10. (Original) The distributed security system of claim 9, wherein the computer device is configured with computer-executable instructions to:

receive from a first entity a message transported with a first transport; and

transmit to the second entity the message using a second transport that is different from the first transport.

11. (Original) The distributed security system of claim 1, wherein the security policy is implemented with at least one application programming interface.

12. (Original) The distributed security system of claim 1, wherein the security language includes programming language constructs.

13. (Original) The distributed security system of claim 1, wherein the security policy includes an identity service.

14. (Original) The distributed security system of claim 1, wherein the security policy includes an admission service.

15. (Original) The distributed security system of claim 1, wherein the security policy includes a permission service.

16. (Original) The distributed security system of claim 1, wherein the security policy includes a revocation service.

17. (Original) The distributed security system of claim 1, wherein the security policy includes a mapping of entities to rights.

18. (Original) The distributed security system of claim 17, wherein the security policy further includes a mapping of entities to capabilities.

19. (Original) The distributed security system of claim 1, wherein the security policy is configured to invoke external computer-readable instructions.

20. (Original) The distributed security system of claim 19, wherein the external computer-readable instructions comprise native processor code.

21. (Original) The distributed security system of claim 19, wherein the external computer-readable instructions comprise Java code.

22. (Withdrawn) A method of delegating security credentials, the method including:

providing to a second party a first license issued to a first party; and

providing to the second party a second license that allows the second party to use the first license.

23. (Withdrawn) The method of claim 22, wherein the second license is issued by the first party.

24. (Withdrawn) The method of claim 22, wherein the second license includes conditions on the use of the first license.

25. (Withdrawn) A method of transmitting a message between a first party and a second party, the method including:

receiving from the first party a message addressed to the second party, wherein the message is transported with a first transport and formatted in accordance with a first protocol;

determining a transport and protocol required by the second party from a security policy; and

transmitting the message to the second party using the transport and protocol required by the second party.

26. (Withdrawn) A method of transmitting a secure message between a first party and a second party, the method including:

formatting the message with a markup language; and

inserting a security credential into a header of the message.

27. (Withdrawn) The method of claim 26, wherein the markup language comprises the extensible markup language.

28. (Withdrawn) The method of claim 26, wherein the security credential comprises a license.

29. (Withdrawn) The method of claim 26, wherein the security credential comprises a key.

30. (Withdrawn) A method of defining a security arrangement between entities of a distributed computing system, the method including:

identifying a portion of a first security policy written in a first security policy language;

identifying a portion of a second security policy written in a second security policy language; and

processing data in accordance with the portion of the first security policy and the portion of the second security policy.

31. (Withdrawn) The method of claim 30, further including exchanging messages between the entities to negotiate on the identification of the portion of the first security policy and the portion of the second security policy.

32. (Withdrawn) The method of claim 30, wherein the first security policy language is the same as the second security policy language.

33. (Previously Presented) A method for utilizing a security policy comprising the steps of:

implementing a security policy written in a security protocol independent security policy language within a distributed computing system, wherein the distributed computing system comprises at least a first computer device operating on a first operating platform and at least a second computer device operating on a second operating platform that supports at least one security protocol that is different than a security protocol supported by a platform of the first computer device; and

configuring the security policy to allow the first computer device and the second computer device to simultaneously process data in accordance with the security policy of the distributed security system.

34. (Previously Presented) A computer readable medium having computer-executable instructions that when executed perform the steps comprising:

applying a security policy within a distributed computing system having at least a first computer device operating on a first operating platform and at least a second computer device operating on a second operating platform that supports at least one security protocol that is different than a security protocol supported by a platform of the first computer device, wherein the security policy is written in a security protocol independent security policy language; and

permitting the security policy to be simultaneously implemented for a plurality of computer devices within the distributed security system including at least the first computer device and the second computer device.

VI. EVIDENCE APPENDIX

None.

VII. RELATED PROCEEDINGS APPENDIX

None.